PROJECT TIME MANAGEMENT

6

Project Time Management includes the processes required to ensure timely completion of the project. Figure 6–1 provides an overview of the following major processes:

6.1 Activity Definition—identifying the specific activities that must be performed to produce the various project deliverables.
6.2 Activity Sequencing—identifying and documenting interactivity dependencies.
6.3 Activity Duration Estimating—estimating the number of work periods which will be needed to complete individual activities.
6.4 Schedule Development—analyzing activity sequences, activity durations, and resource requirements to create the project schedule.
6.5 Schedule Control—controlling changes to the project schedule.

6.1
Activity Definition

6.2
Activity Sequencing

6.3
Activity Duration Estimating

6.4
Schedule Development

6.5
Schedule Control

These processes interact with each other and with the processes in the other knowledge areas as well. Each process may involve effort from one or more individuals or groups of individuals based on the needs of the project. Each process generally occurs at least once in every project phase.

Although the processes are presented here as discrete elements with well-defined interfaces, in practice they may overlap and interact in ways not detailed here. Process interactions are discussed in detail in Chapter 3.

On some projects, especially smaller ones, activity sequencing, activity duration estimating, and schedule development are so tightly linked that they are viewed as a single process (e.g., they may be performed by a single individual over a relatively short period of time). They are presented here as distinct processes because the tools and techniques for each are different.

At present, there is no consensus within the project management profession about the relationship between *activities* and *tasks*:

- In many application areas, activities are seen as being composed of tasks. This is the most common usage and also the preferred usage.
- In others, tasks are seen as being composed of activities.

However, the important consideration is not the term used, but whether or not the work to be done is described accurately and understood by those who must do the work.

6.1 ACTIVITY DEFINITION

Activity definition involves identifying and documenting the specific activities that must be performed in order to produce the deliverables and sub-deliverables identified in the work breakdown structure. Implicit in this process is the need to define the activities such that the project objectives will be met.

Figure 6–1. Project Time Management Overview

Project Time Management

6.1 Activity Definition

- **.1 Inputs**
  - .1 Work breakdown structure
  - .2 Scope statement
  - .3 Historical information
  - .4 Constraints
  - .5 Assumptions
- **.2 Tools and Techniques**
  - .1 Decomposition
  - .2 Templates
- **.3 Outputs**
  - .1 Activity list
  - .2 Supporting detail
  - .3 Work breakdown structure updates

6.2 Activity Sequencing

- **.1 Inputs**
  - .1 Activity list
  - .2 Product description
  - .3 Mandatory dependencies
  - .4 Discretionary dependencies
  - .5 External dependencies
  - .6 Constraints
  - .7 Assumptions
- **.2 Tools and Techniques**
  - .1 Precedence diagramming method (PDM)
  - .2 Arrow diagramming method (ADM)
  - .3 Conditional diagramming methods
  - .4 Network templates
- **.3 Outputs**
  - .1 Project network diagram
  - .2 Activity list updates

6.3 Activity Duration Estimating

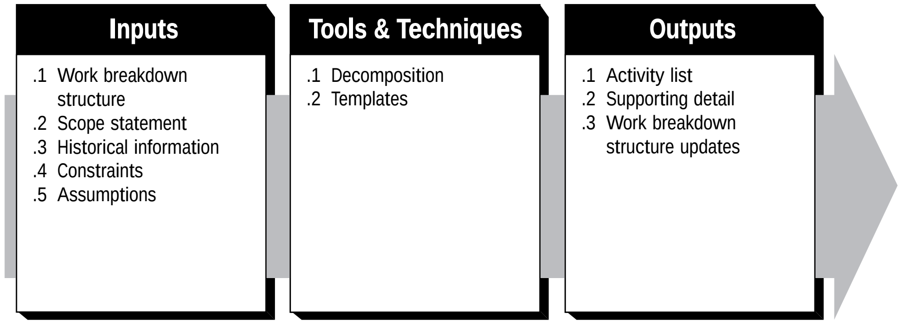
- **.1 Inputs**
  - .1 Activity list
  - .2 Constraints
  - .3 Assumptions
  - .4 Resource requirements
  - .5 Resource capabilities
  - .6 Historical information
- **.2 Tools and Techniques**
  - .1 Expert judgment
  - .2 Analogous estimating
  - .3 Simulation
- **.3 Outputs**
  - .1 Activity duration estimates
  - .2 Basis of estimates
  - .3 Activity list updates

6.4 Schedule Development

- **.1 Inputs**
  - .1 Project network diagram
  - .2 Activity duration estimates
  - .3 Resource requirements
  - .4 Resource pool description
  - .5 Calendars
  - .6 Constraints
  - .7 Assumptions
  - .8 Leads and Lags
- **.2 Tools and Techniques**
  - .1 Mathematical analysis
  - .2 Duration compression
  - .3 Simulation
  - .4 Resource leveling heuristics
  - .5 Project management software
- **.3 Outputs**
  - .1 Project schedule
  - .2 Supporting detail
  - .3 Schedule management plan
  - .4 Resource requirement updates

6.5 Schedule Control

- **.1 Inputs**
  - .1 Project schedule
  - .2 Performance reports
  - .3 Change requests
  - .4 Schedule management plan
- **.2 Tools and Techniques**
  - .1 Schedule change control system
  - .2 Performance measurement
  - .3 Additional planning
  - .4 Project management software
- **.3 Outputs**
  - .1 Schedule updates
  - .2 Corrective action
  - .3 Lessons learned



6.1.1 Inputs to Activity Definition

.1 Work breakdown structure. The work breakdown structure is the primary input to activity definition (see Section 5.3.3.1 for a more detailed discussion of the WBS).

.2 Scope statement. The project justification and the project objectives contained in the scope statement must be considered explicitly during activity definition (see Section 5.2.3.1 for a more detailed discussion of the scope statement).

.3 Historical information. Historical information (what activities were actually required on previous, similar projects) should be considered in defining project activities.

.4 Constraints. Constraints are factors that will limit the project management team's options.

.5 Assumptions. Assumptions are factors that, for planning purposes, will be considered to be true, real, or certain. Assumptions generally involve a degree of risk and will normally be an output of risk identification (described in Section 11.1).

6.1.2 Tools and Techniques for Activity Definition

.1 Decomposition. Decomposition involves subdividing project elements into smaller, more manageable components in order to provide better management control. Decomposition is described in more detail in Section 5.3.2.2. The major difference between decomposition here and in Scope Definition is that the final outputs here are described as activities (actions steps) rather than as deliverables (tangible items). In some application areas, the WBS and the activity list are developed concurrently.

.2 Templates. An activity list (described in Section 6.1.3.1), or a portion of an activity list from a previous project, is often usable as a template for a new project. In addition, the activity list for a WBS element from the current project may be usable as a template for other, similar WBS elements.

6.1.3 Outputs from Activity Definition

.1 Activity list. The activity list must include all activities which will be performed on the project. It should be organized as an extension to the WBS to help ensure that it is complete and that it does not include any activities which are not required as part of the project scope. As with the WBS, the activity list should include descriptions of each activity to ensure that the project team members will understand how the work is to be done.

.2 Supporting detail. Supporting detail for the activity list should be documented and organized as needed to facilitate its use by other project management processes. Supporting detail should always include documentation of all identified assumptions and constraints. The amount of additional detail varies by application area.

.3 *Work breakdown structure updates.* In using the WBS to identify which activities are needed, the project team may identify missing deliverables or may determine that the deliverable descriptions need to be clarified or corrected. Any such updates must be reflected in the WBS and related documentation such as cost estimates. These updates are often called *refinements* and are most likely when the project involves new or unproven technology.

6.2 ACTIVITY SEQUENCING

Activity sequencing involves identifying and documenting interactivity dependencies. Activities must be sequenced accurately in order to support later development of a realistic and achievable schedule. Sequencing can be performed with the aid of a computer (e.g., by using project management software) or with manual techniques. Manual techniques are often more effective on smaller projects and in the early phases of larger ones when little detail is available. Manual and automated techniques may also be used in combination.

Inputs	Tools & Techniques	Outputs
.1 Activity list .2 Product description .3 Mandatory dependencies .4 Discretionary dependencies .5 External dependencies .6 Constraints .7 Assumptions	.1 Precedence Diagramming Method (PDM) .2 Arrow Diagramming Method (ADM) .3 Conditional diagramming methods .4 Network templates	.1 Project network diagram .2 Activity list updates

6.2.1 Inputs to Activity Sequencing

.1 *Activity list.* The activity list is described in Section 6.1.3.1.

.2 *Product description.* The product description is discussed in Section 5.1.1.1. Product characteristics often affect activity sequencing (e.g., the physical layout of a plant to be constructed, subsystem interfaces on a software project). While these effects are often apparent in the activity list, the product description should generally be reviewed to ensure accuracy.

.3 *Mandatory dependencies.* Mandatory dependencies are those which are inherent in the nature of the work being done. They often involve physical limitations (on a construction project it is impossible to erect the superstructure until after the foundation has been built; on an electronics project, a prototype must be built before it can be tested). Mandatory dependencies are also called *hard logic*.

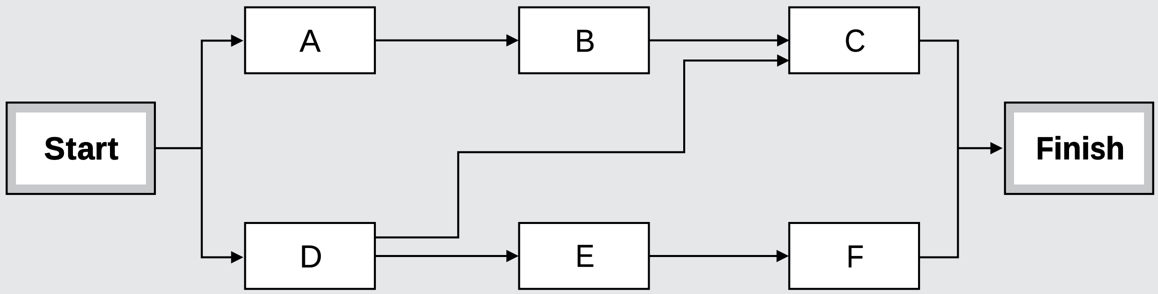
.4 *Discretionary dependencies.* Discretionary dependencies are those which are defined by the project management team. They should be used with care (and fully documented) since they may limit later scheduling options. Discretionary dependencies are usually defined based on knowledge of:

- "Best practices" within a particular application area.
- Some unusual aspect of the project where a specific sequence is desired even though there are other acceptable sequences.

Discretionary dependencies may also be called *preferred logic*, *preferential logic*, or *soft logic*.

.5 *External dependencies.* External dependencies are those that involve a relationship between project activities and non-project activities. For example, the testing activity in a software project may be dependent on delivery of hardware from an exter-

Figure 6–2. Network Logic Diagram Drawn Using the Precedence Diagramming Method



nal source, or environmental hearings may need to be held before site preparation can begin on a construction project.

.6 *Constraints.* Constraints are described in Section 6.1.1.4.

.7 *Assumptions.* Assumptions are described in Section 6.1.1.5.

6.2.2 Tools and Techniques for Activity Sequencing

.1 *Precedence diagramming method (PDM).* This is a method of constructing a project network diagram using nodes to represent the activities and connecting them with arrows that show the dependencies (see also Section 6.2.3.1). Figure 6–2 shows a simple project network diagram drawn using PDM. This technique is also called *activity-on-node* (AON) and is the method used by most project management software packages. PDM can be done manually or on a computer.

It includes four types of dependencies or precedence relationships:

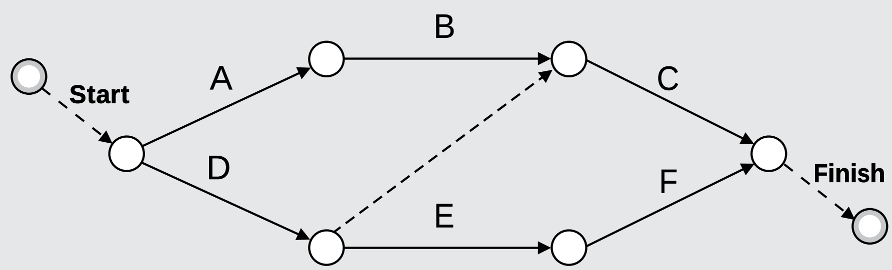
- Finish-to-start—the “from” activity must finish before the “to” activity can start.
- Finish-to-finish—the “from” activity must finish before the “to” activity can finish.
- Start-to-start—the “from” activity must start before the “to” activity can start.
- Start-to-finish—the “from” activity must start before the “to” activity can finish.

In PDM, finish-to-start is the most commonly used type of logical relationship. Start-to-finish relationships are rarely used, and then typically only by professional scheduling engineers. Using start-to-start, finish-to-finish, or start-to-finish relationships with project management software can produce unexpected results since these types of relationships have not been consistently implemented.

.2 *Arrow diagramming method (ADM).* This is a method of constructing a project network diagram using arrows to represent the activities and connecting them at nodes to show the dependencies (see also Section 6.2.3.1). Figure 6–3 shows a simple project network diagram drawn using ADM. This technique is also called *activity-on-arrow* (AOA) and, although less prevalent than PDM, is still the technique of choice in some application areas. ADM uses only finish-to-start dependencies and may require the use of dummy activities to define all logical relationships correctly. ADM can be done manually or on a computer.

.3 *Conditional diagramming methods.* Diagramming techniques such as GERT (Graphical Evaluation and Review Technique) and System Dynamics models allow for non-sequential activities such as loops (e.g., a test that must be repeated more than once) or conditional branches (e.g., a design update that is only needed if the inspection detects errors). Neither PDM nor ADM allow loops or conditional branches.

Figure 6–3. Network Logic Diagram Drawn Using the Arrow Diagramming Method



.4 *Network templates.* Standardized networks can be used to expedite the preparation of project network diagrams. They can include an entire project or only a portion of it. Portions of a network are often referred to as *subnets* or *fragnets*. Subnets are especially useful where a project includes several identical or nearly identical features such as floors on a high-rise office building, clinical trials on a pharmaceutical research project, or program modules on a software project.

6.2.3 Outputs from Activity Sequencing

.1 *Project network diagram.* A project network diagram is a schematic display of the project's activities and the logical relationships (dependencies) among them. Figures 6–2 and 6–3 illustrate two different approaches to drawing a project network diagram. A project network diagram may be produced manually or on a computer. It may include full project details or have one or more summary activities (hammocks). The diagram should be accompanied by a summary narrative that describes the basic sequencing approach. Any unusual sequences should be fully described.

The project network diagram is often incorrectly called a *PERT chart* (for Program Evaluation and Review Technique). A PERT chart is a specific type of project network diagram that is seldom used today.

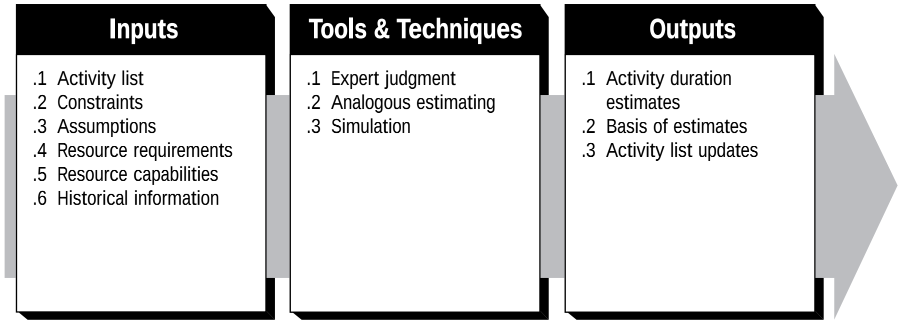
.2 *Activity list updates.* In much the same manner that the activity definition process may generate updates to the WBS, preparation of the project network diagram may reveal instances where an activity must be divided or otherwise redefined in order to diagram the correct logical relationships.

6.3 ACTIVITY DURATION ESTIMATING

Activity duration estimating involves assessing the number of work periods likely to be needed to complete each identified activity. The person or group on the project team who is most familiar with the nature of a specific activity should make, or at least approve, the estimate.

Estimating the number of work periods required to complete an activity will often require consideration of elapsed time as well. For example, if "concrete curing" will require four days of elapsed time, it may require from two to four work periods based on (a) which day of the week it begins on and (b) whether or not weekend days are treated as work periods. Most computerized scheduling software will handle this problem automatically.

Overall project duration may also be estimated using the tools and techniques presented here, but it is more properly calculated as the output of schedule development (described in Section 6.4).



6.3.1 Inputs to Activity Duration Estimating

.1 Activity list. The activity list is described in Section 6.1.3.1.

.2 Constraints. Constraints are described in Section 6.1.1.4.

.3 Assumptions. Assumptions are described in Section 6.1.1.5.

.4 Resource requirements. Resource requirements are described in Section 7.1.3.1. The duration of most activities will be significantly influenced by the resources assigned to them. For example, two people working together may be able to complete a design activity in half the time it takes either of them individually, while a person working half-time on an activity will generally take at least twice as much time as the same person working full-time.

.5 Resource capabilities. The duration of most activities will be significantly influenced by the capabilities of the humans and material resources assigned to them. For example, if both are assigned full-time, a senior staff member can generally be expected to complete a given activity in less time than a junior staff member.

.6 Historical information. Historical information on the likely durations of many categories of activities is often available from one or more of the following sources:

- Project files—one or more of the organizations involved in the project may maintain records of previous project results that are detailed enough to aid in developing duration estimates. In some application areas, individual team members may maintain such records.
- Commercial duration estimating databases—historical information is often available commercially. These databases tend to be especially useful when activity durations are not driven by the actual work content (e.g., how long does it take concrete to cure; how long does a government agency usually take to respond to certain types of requests).
- Project team knowledge—the individual members of the project team may remember previous actuals or estimates. While such recollections may be useful, they are generally far less reliable than documented results.

6.3.2 Tools and Techniques for Activity Duration Estimating

.1 Expert judgment. Expert judgment is described in Section 5.1.2.2. Durations are often difficult to estimate because of the number of factors which can influence them (e.g., resource levels, resource productivity). Expert judgment guided by historical information should be used whenever possible. If such expertise is not available, the estimates are inherently uncertain and risky (see Chapter 11, Project Risk Management).

.2 *Analogous estimating.* Analogous estimating, also called *top-down estimating*, means using the actual duration of a previous, similar activity as the basis for estimating the duration of a future activity. It is frequently used to estimate project duration when there is a limited amount of detailed information about the project (e.g., in the early phases). Analogous estimating is a form of expert judgment (described in Section 6.3.2.1).

Analogous estimating is most reliable when (a) the previous activities are similar in fact and not just in appearance, and (b) the individuals preparing the estimates have the needed expertise.

.3 *Simulation.* Simulation involves calculating multiple durations with different sets of assumptions. The most common is Monte Carlo Analysis in which a distribution of probable results is defined for each activity and used to calculate a distribution of probable results for the total project (see also Section 11.2.2.3, Schedule Simulation).

6.3.3 Outputs from Activity Duration Estimating

.1 *Activity duration estimates.* Activity duration estimates are quantitative assessments of the likely number of work periods that will be required to complete an activity.

Activity duration estimates should always include some indication of the range of possible results. For example:

- 2 weeks ± 2 days to indicate that the activity will take at least 8 days and no more than 12.
- 15 percent probability of exceeding 3 weeks to indicate a high probability—85 percent—that the activity will take 3 weeks or less.

Chapter 11 on Project Risk Management includes a more detailed discussion of estimating uncertainty.

.2 *Basis of estimates.* Assumptions made in developing the estimates must be documented.

.3 *Activity list updates.* Activity list updates are described in Section 6.2.3.2.

6.4 SCHEDULE DEVELOPMENT

Schedule development means determining start and finish dates for project activities. If the start and finish dates are not realistic, the project is unlikely to be finished as scheduled. The schedule development process must often be iterated (along with the processes that provide inputs, especially duration estimating and cost estimating) prior to determination of the project schedule.

Inputs	Tools & Techniques	Outputs
.1 Project network diagram .2 Activity duration estimates .3 Resource requirements .4 Resource pool description .5 Calendars .6 Constraints .7 Assumptions .8 Leads and Lags	.1 Mathematical analysis .2 Duration compression .3 Simulation .4 Resource leveling heuristics .5 Project management software	.1 Project schedule .2 Supporting detail .3 Schedule management plan .4 Resource requirement updates

6.4.1 Inputs to Schedule Development

.1 *Project network diagram.* The project network diagram is described in Section 6.2.3.1.

.2 *Activity duration estimates.* Activity duration estimates are described in Section 6.3.3.1.

.3 *Resource requirements.* Resource requirements are described in Section 6.3.1.4.

.4 *Resource pool description.* Knowledge of what resources will be available at what times and in what patterns is necessary for schedule development. For example, shared resources can be especially difficult to schedule since their availability may be highly variable.

The amount of detail and the level of specificity in the resource pool description will vary. For example, for preliminary schedule development of a consulting project one need only know that two consultants will be available in a particular timeframe. The final schedule for the same project, however, must identify which specific consultants will be available.

.5 *Calendars.* Project and resource calendars identify periods when work is allowed. *Project calendars* affect all resources (e.g., some projects will work only during normal business hours while others will work a full three shifts). *Resource calendars* affect a specific resource or category of resources (e.g., a project team member may be on vacation or in a training program; a labor contract may limit certain workers to certain days of the week).

.6 *Constraints.* Constraints are described in Section 6.1.1.4. There are two major categories of constraints that must be considered during schedule development:

- Imposed dates. Completion of certain deliverables by a specified date may be *required* by the project sponsor, the project customer, or other external factors (e.g., a market window on a technology project; a court-mandated completion date on an environmental remediation project).
- Key events or major milestones. Completion of certain deliverables by a specified date may be *requested* by the project sponsor, the project customer, or other stakeholders. Once scheduled, these dates become expected and often may be moved only with great difficulty.

.7 *Assumptions.* Assumptions are described in Section 6.1.1.5.

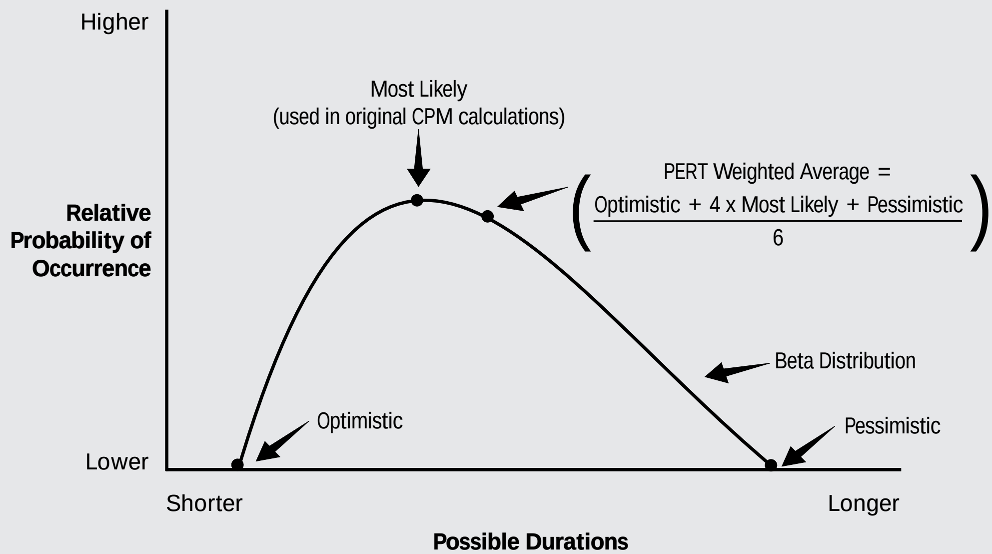
.8 *Leads and lags.* Any of the dependencies may require specification of a lead or a lag in order to accurately define the relationship (e.g., there might be a two-week delay between ordering a piece of equipment and installing or using it).

6.4.2 Tools and Techniques for Schedule Development

.1 *Mathematical analysis.* Mathematical analysis involves calculating theoretical early and late start and finish dates for all project activities without regard for any resource pool limitations. The resulting dates are not the schedule, but rather indicate the time periods within which the activity *should* be scheduled given resource limits and other known constraints. The most widely known mathematical analysis techniques are:

- Critical Path Method (CPM)—calculates a single, deterministic early and late start and finish date for each activity based on specified, sequential network logic and a single duration estimate. The focus of CPM is on calculating *float* in order to determine which activities have the least scheduling flexibility. The underlying CPM algorithms are often used in other types of mathematical analysis.
- Graphical Evaluation and Review Technique (GERT)—allows for probabilistic treatment of both network logic and activity duration estimates (i.e., some activities may not be performed at all, some may be performed only in part, and others may be performed more than once).
- Program Evaluation and Review Technique (PERT)—uses sequential network logic and a weighted average duration estimate to calculate project duration. Although there are surface differences, PERT differs from CPM primarily in that it uses the distribution's mean (expected value) instead of the most likely estimate originally used in CPM (see Figure 6–4). PERT itself is seldom used today although PERT-like estimates are often used in CPM calculations.

Figure 6–4. PERT Duration Calculation



.2 *Duration compression.* Duration compression is a special case of mathematical analysis that looks for ways to shorten the project schedule without changing the project scope (e.g., to meet imposed dates or other schedule objectives). Duration compression includes techniques such as:

- Crashing—in which cost and schedule trade-offs are analyzed to determine how to obtain the greatest amount of compression for the least incremental cost. Crashing does not always produce a viable alternative and often results in increased cost.
- Fast tracking—doing activities in parallel that would normally be done in sequence (e.g., starting to write code on a software project before the design is complete, or starting to build the foundation for a petroleum processing plant before the 25 percent of engineering point is reached). Fast tracking often results in rework and usually increases risk.

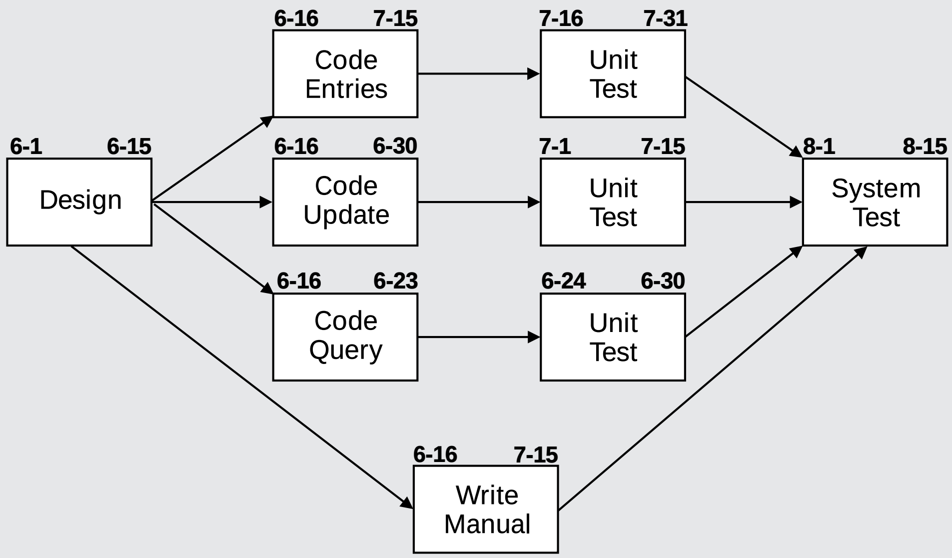
.3 *Simulation.* Simulation is described in Section 6.3.2.3.

.4 *Resource leveling heuristics.* Mathematical analysis often produces a preliminary schedule that requires more resources during certain time periods than are available, or requires changes in resource levels that are not manageable. Heuristics such as “allocate scarce resources to critical path activities first” can be applied to develop a schedule that reflects such constraints. Resource leveling often results in a project duration that is longer than the preliminary schedule. This technique is sometimes called the “Resource-based Method,” especially when implemented with computerized optimization.

Resource constrained scheduling is a special case of resource leveling where the heuristic involved is a limitation on the quantity of resources available.

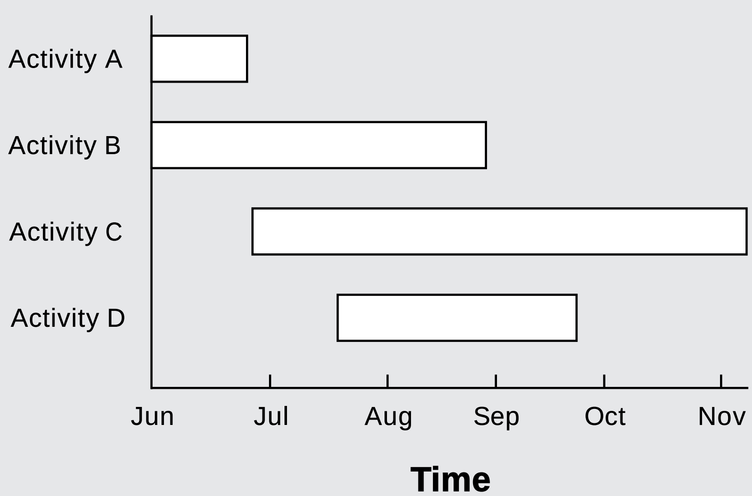
.5 *Project management software.* Project management software is widely used to assist with schedule development. These products automate the calculations of mathematical analysis and resource leveling and thus allow for rapid consideration of many schedule alternatives. They are also widely used to print or display the outputs of schedule development.

Figure 6–5. Project Network Diagram with Scheduled Dates



There are many other acceptable ways to display date information on a project network diagram. This figure shows start and finish dates without time-of-day information.

Figure 6–6. Bar (Gantt) Chart

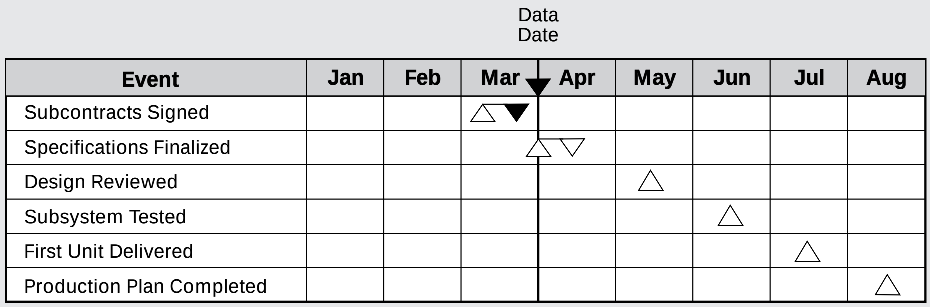


There are many other acceptable ways to display project information on a bar chart.

6.4.3 Outputs from Schedule Development

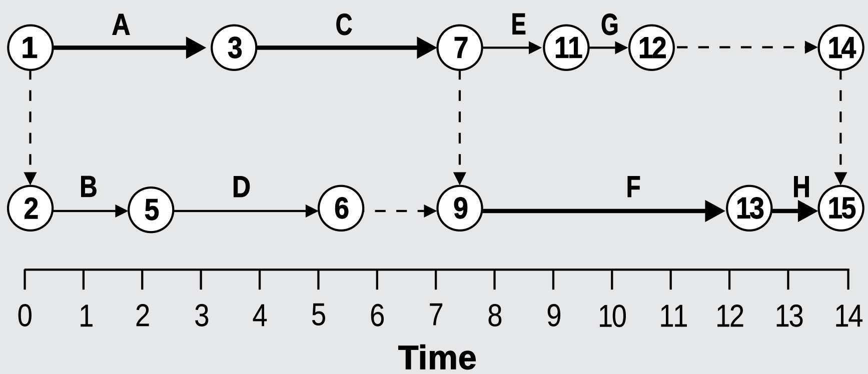
.1 *Project schedule.* The project schedule includes at least planned start and expected finish dates for each detail activity. (Note: the project schedule remains preliminary until resource assignments have been confirmed. This would usually happen no later than the completion of Project Plan Development, Section 4.1).

Figure 6–7. Milestone Chart



There are many other acceptable ways to display project information on a milestone chart.

Figure 6–8. Time-Scaled Network Diagram



There are many other acceptable ways to display project information on a time-scaled network diagram.

The project schedule may be presented in summary form (the “master schedule”) or in detail. Although it can be presented in tabular form, it is more often presented graphically using one or more of the following formats:

- Project network diagrams with date information added (see Figure 6–5). These charts usually show both the project logic and the project’s critical path activities (see Section 6.2.3.1 for more information on project network diagrams).
- Bar charts, also called Gantt charts (see Figure 6–6), show activity start and end dates as well as expected durations, but do not usually show dependencies. They are relatively easy to read and are frequently used in management presentations.
- Milestone charts (see Figure 6–7), similar to bar charts, but identifying the scheduled start or completion of major deliverables and key external interfaces.
- Time-scaled network diagrams (see Figure 6–8) are a blend of project network diagrams and bar charts in that they show project logic, activity durations, and schedule information.

.2 *Supporting detail.* Supporting detail for the project schedule includes at least documentation of all identified assumptions and constraints. The amount of additional detail varies by application area. For example:

- On a construction project, it will most likely include such items as resource histograms, cash flow projections, and order and delivery schedules.
- On an electronics project, it will most likely include resource histograms only.

Information frequently supplied as supporting detail includes, but is not limited to:

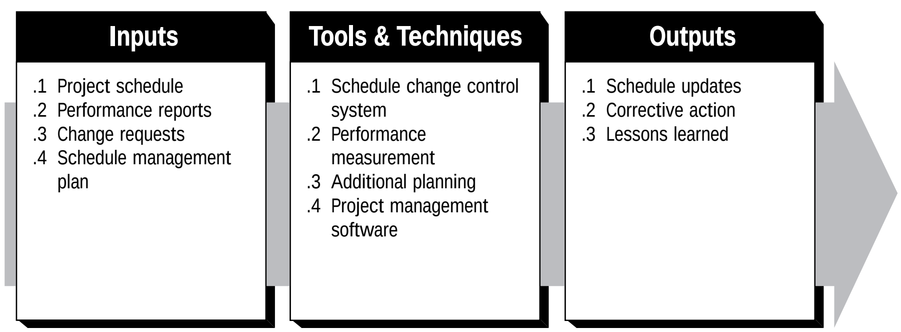
- Resource requirements by time period, often in the form of a resource histogram.
- Alternative schedules (e.g., best case or worst case, resource leveled or not, with or without imposed dates).
- Schedule reserves or schedule risk assessments (see Section 11.3.3).

.3 *Schedule management plan.* A schedule management plan defines how changes to the schedule will be managed. It may be formal or informal, highly detailed or broadly framed based on the needs of the project. It is a subsidiary element of the overall project plan (see Section 4.1).

.4 *Resource requirement updates.* Resource leveling and activity list updates may have a significant effect on preliminary estimates of resource requirements.

6.5 SCHEDULE CONTROL

Schedule control is concerned with (a) influencing the factors which create schedule changes to ensure that changes are beneficial, (b) determining that the schedule has changed, and (c) managing the actual changes when and as they occur. Schedule control must be thoroughly integrated with the other control processes as described in Section 4.3, Overall Change Control.



6.5.1 Inputs to Schedule Control

.1 *Project schedule.*The project schedule is described in Section 6.4.3.1. The approved project schedule, called the schedule baseline, is a component of the overall project plan described in Section 4.1.3.1. It provides the basis for measuring and reporting schedule performance.

.2 *Performance reports.* Performance reports, discussed in Section 10.3.3.1, provide information on schedule performance such as which planned dates have been met and which have not. Performance reports may also alert the project team to issues which may cause problems in the future.

.3 *Change requests.* Change requests may occur in many forms—oral or written, direct or indirect, externally or internally initiated, and legally mandated or optional. Changes may require extending the schedule or may allow accelerating it.

.4 *Schedule management plan.* The schedule management plan is described in Section 6.4.3.3.

6.5.2 Tools and Techniques for Schedule Control

.1 *Schedule change control system*. A schedule change control system defines the procedures by which the project schedule may be changed. It includes the paperwork, tracking systems, and approval levels necessary for authorizing changes. Schedule change control should be integrated with the overall change control system described in Section 4.3.

.2 *Performance measurement*. Performance measurement techniques such as those described in Section 10.3.2 help to assess the magnitude of any variations which do occur. An important part of schedule control is to decide if the schedule variation requires corrective action. For example, a major delay on a non-critical activity may have little effect on the overall project while a much shorter delay on a critical or near-critical activity may require immediate action.

.3 *Additional planning.* Few projects run exactly according to plan. Prospective changes may require new or revised activity duration estimates, modified activity sequences, or analysis of alternative schedules.

.4 *Project management software.* Project management software is described in Section 6.4.2.5. The ability of project management software to track planned dates versus actual dates and to forecast the effects of schedule changes, real or potential, makes it a useful tool for schedule control.

6.5.3 Outputs from Schedule Control

.1 *Schedule updates.* A schedule update is any modification to the schedule information which is used to manage the project. Appropriate stakeholders must be notified as needed. Schedule updates may or may not require adjustments to other aspects of the overall project plan.

*Revision*s are a special category of schedule updates. Revisions are changes to the scheduled start and finish dates in the approved project schedule. These dates are generally revised only in response to scope changes. In some cases, schedule delays may be so severe that “rebaselining” is needed in order to provide realistic data to measure performance.

.2 *Corrective action.* Corrective action is anything done to bring expected future schedule performance into line with the project plan. Corrective action in the area of time management often involves expediting: special actions taken to ensure completion of an activity on time or with the least possible delay.

.3 *Lessons learned.* The causes of variances, the reasoning behind the corrective action chosen, and other types of lessons learned from schedule control should be documented so that they become part of the historical database for both this project and other projects of the performing organization.